international journal of critical infrastructure protection 35 (2021) 100466

Contents lists available at ScienceDirect

International Journal of Critical Infrastructure Protection

journal homepage: www.elsevier.com/locate/ijcip

Threat Modelling of Cyber–Physical Systems Using an Applied π-Calculus

Livinus Obiora Nweke [a,*], Goitom K. Weldehawaryat [a], Stephen D. Wolthusen [a,b]

[a] Norwegian University of Science and Technology (NTNU), Gjøvik, Norway
[b] Royal Holloway, University of London, Egham, United Kingdom

ARTICLE INFO

Keywords:
Threat modelling
Cyber–physical systems
π-Calculus
Attack–Defence Trees
Partial ordering

ABSTRACT

Cyber–Physical Systems (CPS) are distributed systems in which the state of the physical system is generally not observable in non-trivial cases, and where state transitions of this physical system can also occur without resulting in immediate changes to observable variables. This poses challenges for the bidirectional synchronisation of the discrete cyber models and the partially continuous physical systems. Threats to CPS from cyber attacks are, however, often instantiable only where conditions on the CPS state during the attack meet certain conditions such that they drive the system state outside a desirable or safe space.

In this paper we propose an extension to an applied π-calculus in which we can capture both the behaviour of the CPS as well as modelling possible adversary behaviour. This is achieved by embedding an algebraic representation of Attack–Defence Trees (ADT) in the applied π-calculus and augmenting this by the addition of a partial ordering over the constituents of the ADT within the embedding, offering an elegant mechanism to extend ADT to ordering and time-related attacks. We illustrate the modelling approach for the case of an electrical substation fragment in which components communicate via the IEC 61850 protocol.

1. Introduction

Attacks against CPS are no longer theoretical concepts but are real and here before us. The attack against Natanz uranium enrichment plant in Iran, where the infamous malware known as Stuxnet escaped the digital realm and wreaked physical damage to a CPS [1] attest to that fact. Most recently, researchers have discovered Ekans ransomware, which they observed was specifically designed to target CPS [2]. These trends call for rethinking about how we threat model a CPS considering timing, uncertainty, and dependencies that exist between its entities.

Threat modelling approaches provide a systematic way of reasoning about the potential threats to a system. Threats are events that could compromise the confidentiality, integrity, or availability of a system, through unauthorised disclosure, misuse, alteration or destruction [3]. There are three main approaches that may be employed to threat model a system and they include: approaches that are concerned with the assets of the system being threat modelled (asset-centric threat modelling); approaches that aim to understand the nature of the attackers (attack-centric threat modelling); and approaches that focus on the software or the system (software-centric or system-centric threat modelling) [4,5]. We are interested in using an applied π-calculus to capture both the behaviour of the CPS as well as modelling possible adversary behaviour.

CPS integrate both computation and communication capabilities in order to control physical components. The computational elements are used for processing measurement values received from the physical components through the communication channels. This entails that there are several assets that make up the CPS and the use of threat modelling approach would be effective for understanding the potential threats to CPS. However, the existing threat modelling approaches are not able to capture the potential threats to CPS due to the timing, uncertainty, and dependencies that exist between the entities of CPS.

In a CPS, the measurement values obtained from the physical components are used to ascertain the state of the system. Unless these parameters and how they interact with one another are threat modelled, it will be difficult to know how threats to these assets may be exploited. For example, CPS may have a handful of critical states and if an attack is launched when the system is not in a critical state, the impact may not be adverse. However, an attack that is launched when the system is in a critical state would have a devastating effect. The question that is important for a threat modelling approach to consider is then; what is the likelihood of an attacker finding the system in a critical state in order to launch an attack to obtain an adverse impact.

The assertion we are making in this paper is that the likelihood of an attacker finding a CPS in a critical state cannot be expressed using the existing threat modelling approaches. The big risk of a more abstract

* Corresponding author.
E-mail address: livinus.nweke@ntnu.no (L.O. Nweke).

https://doi.org/10.1016/j.ijcip.2021.100466
Received 1 December 2020; Received in revised form 17 April 2021; Accepted 14 July 2021
Available online 22 July 2021

approach like the existing threat modelling approaches is that there are chances of making a mistake by either being too confident that the approach is going to find the threats even though an understanding of the critical processes or states is missed or wasting a lot of resources by defending an attack that is very unlikely. For instance, the processes in OCTAVE (operationally threat asset and vulnerability evaluation) [6] are narrative driven. They consider the assets of a system and the interaction between these assets, then employ verbal reasoning to evaluate the threats to the systems. We argue that this type of intuitive reasoning no longer suffices for CPS where timing, uncertainty, and dependencies between the entities exist.

Hence, we extend the attack–defence trees (ADT) with partial ordering to represent causality relationship and use an applied π-calculus to describe the formal model for CPS, taking into account its unique properties. We then define the semantics of the applied π-calculus using a labelled transition system to highlight the CPS interactions with the environment and to facilitate the definition of observational equivalences such as *bisimilarity*. This is to allow us to capture the potential threats to the CPS and to deduce some reasoning about the behaviour of the system. Also, to show the utility of our model, we present a use case scenario where the applied π-calculus is employed to reason about false measurement injection attack against IEC 61850 protocol. The main contributions of this paper are as follows.

- We propose an extension to an applied π-calculus in which we can capture both the behaviour of the CPS as well as modelling possible adversary behaviour.
- We use the ADT and extend it with partial ordering of events to show causality relationship. This allows us to represent not only the necessary conditions for an attack to be successful but also the sequencing of events or the way events have to be order for the attack to be successful.
- We translate the ADT with partial ordering into the applied π-calculus using the message synchronisation primitives for partial ordering, which enables us to make an argument for equivalence.
- We illustrate our modelling approach for the case of an electrical substation fragment in which components communicate via the IEC 61850 protocol.

The rest of this paper is organised as follows. Section 2 presents a discussion on CPS, where we use smart grid system as an example of CPS. Also, we briefly describe the π-calculus that will be employed to threat model the CPS. Section 3 discusses the different approaches that have been utilised in the literature to threat model CPS. Section 4 provides justification on why the existing threat modelling approaches are not able to capture the unique properties of a CPS. Section 5 presents the formal model for a CPS using an applied π-calculus and threat modelling of the CPS based on the formal model. In addition, the section describes a use case scenario to show the utility of our model. Section 6 concludes the paper and presents future works.

2. Background

In this section, we present a discussion on CPS. We use smart grid system as an example of CPS to describe a high-level architecture, where we extract some properties that are specific to CPS. Also, we briefly discuss π-calculus that will be employed to threat model the CPS.

2.1. Cyber–physical systems

CPS are systems that consist of computation, communication, and physical components. They combine computing and communication capabilities with the monitoring and control of assets in the physical domain [7]. Some of these systems are usually referred to as real-time systems with stringent quality of service (QoS) requirements. Also, the coupling of physical and cyber components entails that any malicious activity in the cyber components would have devastating effects on the physical components, which in turn may endanger the lives of humans operating the physical components. For this reason, CPS are sometimes called safety-critical systems. The application of CPS span through several domains including; power stations, power and water distribution, traffic systems, oil and gas sector, etc.

In CPS, there are several assets that makeup the system and they can be classified as follows: the cyber and control part assets, the physical assets, and the communication channel between the cyber and physical assets. The cyber assets consist of hardware, software, and data that connects to the Internet infrastructure. For the physical assets, they include sensors and actuators that monitor the physical environment. And the communication channels are assets used to send data from the physical environment to the cyber and control parts, and commands from the cyber and control parts to the sensors and actuators. A high-level description of these assets and the communication between them is shown in Fig. 1.

Fig. 1 depicts a smart grid system which is an example of a complex CPS. In a smart grid system, the conventional electrical grid has been integrated with information communication technology (ICT). Smart grid system consists of several assets, and they can be classified as we have already done in the preceding paragraph. The cyber and control part assets include the supervisory control and data acquisition (SCADA) which facilitates the interconnection of the field devices like the sensors, actuator, etc. Also, the communication asset (Communication Network) that ensures bidirectional communication of data and signals in the smart grid, in addition to enabling interaction between the cyber and physical assets. Lastly, the physical assets which include the transformers, power transmission networks, distribution networks, etc.

2.2. π-Calculus

The π-calculus is a process algebra proposed by Robin Milner [9] for describing and analysing concurrent systems with evolving communication structure. It provides a formal mechanism for modelling communication among processes over dynamic links [10] and has since been extended and applied in several studies including for modelling different types of security processes [11–14]. A system in the π-calculus is made up of independent processes that communicate via channels. A channel is an abstraction of the communication link and is referred to by name. Names are the simplest entities of the π-calculus and there are infinite number of names, represented by lowercase letters (x, y, z, etc.).

Processes in the π-calculus evolve by performing actions. These capabilities for action are expressed via the prefixes, of which there are four kinds:

$$\pi := \overline{x}y \mid x(z) \mid \tau \mid [x = y]\pi$$

The first capability is to send the name y via name x, and the second to receive any name via x. The third capability refers to internal action or unobservable action. And lastly, the fourth is a conditional capability where the capability π is executed if x and y are the same. The set of processes can also be defined by the syntax given in Table 1.

- A composition $P|Q$ behaves as if processes P and Q are running in parallel. This implies that the two processes can evolve separately at the same time and can operate on the channels to communicate with each other and with the outside the network.
- The basic interaction is defined using $\overline{x}z.P$ that defines an output process that is ready to output on channel x, or $x(y).P$ that defines an input process that is ready to receive a value over channel x.
- The replication $!P$ behaves as an infinite number of copies of P running in parallel.
- The name restriction operator $(\nu x.P)$ is a process that makes a new, private name x, and then behaves as P.

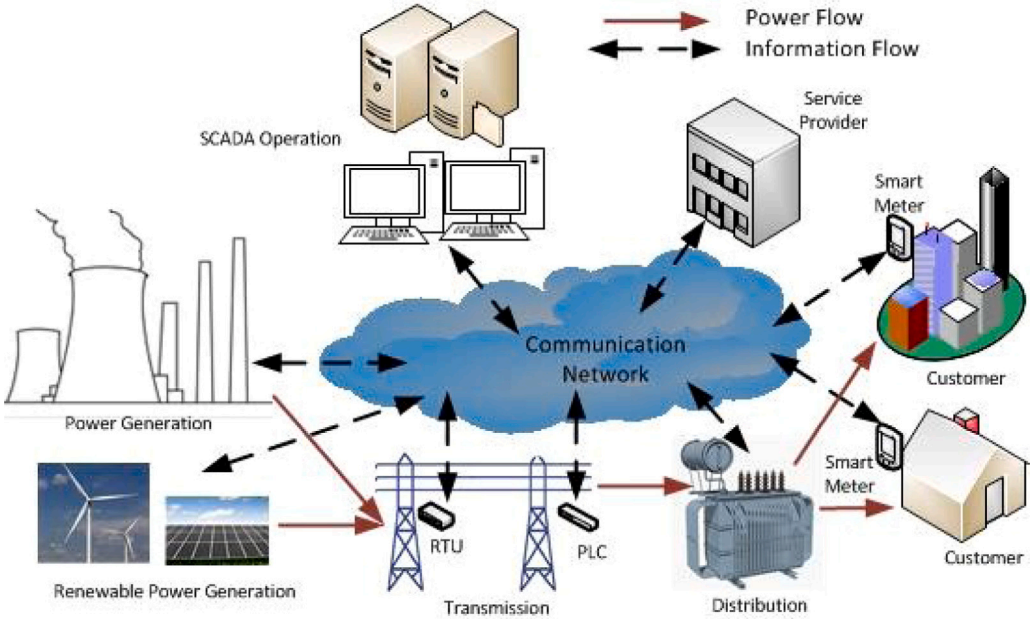

Fig. 1. Smart grid system [8].

Table 1
Syntax of π-calculus.

Term	Semantics
$P ::=$	Processes
0	Empty process
$\overline{x}z.P$	Output
$x(y).P$	Input
$P + Q$	Choice
$P\|Q$	Parallel composition
$!P$	Replication
$\nu x.P$	Restriction
τ	Silent function/action

- τ represents the internal (silent) action of a process that is not observable outside the scope of the process.
- 0 is the empty process.

To briefly describe the use of the π-calculus for modelling systems, let consider the following example which is similar to the illustration provided by Parrow in [15]. Suppose we have a system which consists of three processes, namely: a controller, a resource and an agent. The controller controls access to the resource and the agent needs access to it. We can represent the original state of the controller using a communication link x. The agent interacts with the controller via another link y to have access to the resource. After this interaction, access to the resource will be transferred to the agent. We can express the communication among these processes using the π-calculus as follows: the controller that sends x along y is $\overline{y}x.C$; the agent that receives some link along y and then uses it to send data along it is $y(a).\overline{a}z.A$. The interaction we have described so far can be formulated in the π-calculus as follows:

$$\overline{y}x.C \mid y(a).\overline{a}z.A \xrightarrow{\tau} C \mid \overline{x}z.A$$

However, we use the extended pi-calculus in this work to model CPS. As we have observed already, CPS consist of a physical component that embodies all physical aspects of a system (state variables, physical devices, etc.) and a cyber component that interacts with the physical devices (sensors and actuators) of the system and can communicate via channels with other processes of the same CPS or other CPS. The overall behaviour of the CPS is structured by the combination of the behaviours of its subsystems. Thus, in Section 5 we use the capability of the applied π-calculus to model the message exchanges/interactions that captures the specific behaviour associated with a CPS.

3. Related works

CPS security has generated a lot of attention in recent years. A significant amount of research effort has been dedicated towards the analysis, detection and identification of security issues in CPS. For example, Mo et al. [16] develop a model-based techniques capable of detecting integrity attacks on the sensors of a control system. Also, Pasqualetti et al. in [17] present attack detection and identification in CPS and analyse the core monitoring limitations for CPS under attack modelled by linear time-invariant descriptor systems with exogenous inputs. Several other works have considered security issues in CPS, such as denial-of-service attacks [18,19], replay attacks [20–22], and false data injection attacks [23–25]. However, the threat model used in most of these works employs custom construct which makes them difficult to use in different environments. Our proposed model offers a set of constructs that can be used to decompose threats in CPS.

Threat modelling of CPS has been attempted in several works in the literature. One of the earliest attempts to threat model CPS came from Zalewski et al. [26]. They propose the use of a discrete time Markov chain (DTMC) model to characterise the transitions between the secure and insecure states of CPS. The authors argue that quantifying the probabilities of transitions between secure and insecure states will allow for the derivation of important inferences about the security related features of CPS. Then, the conventional threat modelling techniques (STRIDE, DREAD, CVSS) are applied in the work, to assign the probabilities of transitions between the states. These techniques capture threats at certain level abstraction which does not allow for reasoning over the communication between assets and their timing property.

Martins et al. in [27] present a tool to perform a systematic threat modelling for CPS using a real-world temperature monitoring system as a case study. The authors use the Generic Modelling Environment for the creation of domain-specific modelling for threat analysis CPS. Also, they extended and deployed Microsoft SDL Threat Modelling Tools to model, identify, and mitigate threats in a systematic way for the proposed CPS. A model to represent CPS threats using patterns that are related to architectural aspects of the CPS is described in [7]. The author shows how to extend the misuse pattern to characterise

cyber–physical threats and how to enumerate and unify cyber–physical threats.

A threat modelling framework for CPS using STRIDE is presented in [28]. The authors demonstrate the applicability of the proposed framework using a real synchrophasor-based synchronous islanding test-bed in the laboratory. They show that an adversary can achieve a specific malicious goal by exploiting threats at different locations in the system. Also, they illustrate that by identifying component level vulnerabilities and their potential physical consequence, STRIDE can be applied to address such challenge. Almohri et al. in [29] present threat modelling of medical CPS. The authors consider the roles of stakeholders and system components. They use this understanding to sketch an abstract architecture of medical CPS and then show the various threat modelling options.

CPS threats and vulnerabilities analysis for train control and monitoring systems is presented in [30]. The authors evaluate vulnerabilities and characteristics of railway threat landscape including potential threats, threat actors and motivations. Also, they examine the direct impacts and cascading consequences of threats on the whole system as well as risk produced. Atif et al. in [31] describe cyber threat analysis for CPS. They employ data-driven approach to threat model CPS. A machine learning algorithm based on K-Nearest-Neighbour (K-NN) is used in this work, to ascertain the threat category faced by the CPS considered.

Attacker models for CPS have been discussed in [32]. The authors present a literature review of the attacker models for CPS and define a taxonomy of ten different features that they applied to the literature. Also, a generalised attacker and attack models for CPS has been proposed in [33] and have been employed to investigate the impact of single-point cyber attacks on a Secure Water Treatment (SWaT) system in [34]. Unlike these attacker and attack models presented in [32–34] where the authors utilise descriptive threat modelling techniques, our approach allows us to be analytical. We are able to describe the adversary behaviour at a level of details that allows us to effectively explore the range of parameters or the behaviour of a system. This enables us to infer not only the necessary conditions for an attack to be successful but also the sequencing of events or the way events have to be order for the attack to be successful.

In contrast to all the works described in this section, we present threat modelling of CPS using an applied π-calculus in this paper. The applied calculus has also been used for attacking modelling in [14]. The main differentiator of our approach to the existing literature on threat modelling and attack modelling in CPS is two fold. First, our method has the potential to be automated. This is because it is possible to take an applied pi-calculus model and translate it into a theorem prover or prover assistance and then perform the reasoning automatically. However, it requires that the specification is sufficiently precise that it can be used to reason over the semantics. Second, our approach allows us to analyse the threats to CPS in a more precise way. It enables us to capture the pre-conditions that are applicable to certain types of threats. This is because a CPS will not always be vulnerable: there will be some states where manipulating a variable will have an effect and there will be other states where manipulating the same variable will not have an effect. Our method allows us to represent these states in the form of processes and the interaction between these processes and to reason about the likelihood of an attacker finding the system in a critical state to launch an attack for adverse effect.

4. Towards threat modelling of cyber–physical systems using an applied π-calculus

We conducted a review of the existing asset-centric threat modelling approaches in [35] and observed that the intuitive reasoning approach employed in those threat modelling approaches is not sufficient to threat model CPS where uncertainty, timing and dependencies between the entities exist. In threat modelling of CPS, we have to take into account that the processes we are trying to capture are not all predictable and deterministic [36]. Also, Murphy's law holds in CPS as it has been noted that a system with vulnerabilities will be exploited given a suitable operational environment [37]. A possible corollary for Murphy's law in CPS is that because of some variations in the process, one would obtain abnormal behaviour in the CPS at the same time as someone probing or attacking the system.

The existing threat modelling approaches assume that the systems being threat modelled are in a normal state when the attacker might strike. This type of assumption cannot capture all the possible threat scenarios in CPS because there are likelihoods that something may be going wrong with the CPS and at the same time a threat may be stressing the system even more. Situations like this are unlikely to occur in conventional systems, where the existing threat modelling approaches are usually applied. Hence, we need to find a way of representing the interaction that are occurring between entities in CPS taking into account the unpredictable and nondeterministic behaviour of CPS.

Moreover, the existing threat modelling approaches are not good at expressing timing property between assets and operations. CPS as we have already observed, may have a handful of critical states. The threat modelling process for CPS should have a way of depicting such critical states because if an attack is launched when CPS is not in a critical state, the impact may be negligible. Unfortunately, the existing threat modelling approaches do not have a way of expressing the likelihood that an attacker may find the system in a critical state. Thus, it is important to consider an appropriate threat modelling approach for CPS, which takes into consideration the timing property between assets and operations.

Also, the inherent nature of CPS implies that there are dependencies between the assets at different levels and the operations of the system. It is no longer the case that the behaviour of the system can be understood by looking at the assets at the different components in isolation, but rather in combination with other assets. This is because an asset in CPS may be critical not in its own right, but instead as a provider for services in another asset. Also, these dependencies can be annotated with additional requirements to reason about how threats to these assets may be realised.

In additional to the dependencies between the assets at different levels and the operations of CPS, threats to availability are important requirements to consider. In a typical CPS, we are dealing with availability problem for example, redundancy. We are interested in expressing risks and threats to assets and services which can be provided in different ways. It is possible to examine a situation where we have an asset with a vulnerability, and to know if we can replace the output of that asset with some other substitute asset to give the same input to another asset that depends on the asset with a vulnerability.

By explicitly exposing the communication between the assets of a CPS using an applied π-calculus, we can deduce some reasoning about the behaviour of the system. One of the things that could be interesting for such a model is what it tells us about, for example, what an adversary can and cannot know about the state of CPS. There are some assets of the system that are only going to expose some information through messages or other interactions; and if an attacker is place at a particular place in that topology, the attacker would not be able to see any interaction unless there is a way of getting the message across. This is a useful insight because it might mean that any adversary would not be able to make use of this information and may have to rely on some sort of model.

Generally, we need a formal way of expressing these requirements that are peculiar to CPS environment. This will facilitate the deployment of an appropriate threat modelling approach for the identification of threats to assets in CPS. So far, different methodologies have been proposed for formal modelling of CPS for the purpose of identifying threats to the system. However, we employ an applied π-calculus in this paper as described in following section to formally model the CPS environment. Then, we deploy the threat modelling approach using an applied π-calculus to evaluate threats that are applicable to CPS.

5. Threat modelling of cyber–physical systems using an applied π-calculus

In this section, we present a formal model for CPS using an applied π-calculus. The formal model takes into account the requirements identified in the preceding subsection to ensure that the threats to CPS are captured. We then use the formal model to threat modelled the CPS.

5.1. Formal model for cyber–physical system

An essential step in threat modelling a system is to develop a model of the system to be threat modelled. This would allow for the identification assets of the system and to reason about the likelihood of those assets being compromised. The process of identifying assets of a system is an important approach in threat analysis. It provides the security practitioner with insights into the most critical assets of the system and ensures efficient deployment of resources to protect those critical assets.

A CPS has a physical process under its control, a set of sensors that report the state of the process to a controller, which in turn sends control signals to actuators to maintain the system in a desired state. The controller also communicates with a supervisory and configuration device (e.g., a SCADA system in the power grid) which can monitor the system or change the settings of the controller. Fig. 1 illustrates an example of CPS architecture. CPS consist of two components: a *physical plant/environment* that encloses all physical aspects of a system (state variables, physical devices, etc.) and a *cyber component* represented as a concurrent process that interacts with the physical devices (sensors and actuators) of the system and can communicate via channels with other processes of the same CPS or with processes of other CPS.

CPS are widely modelled as a linear discrete-time stochastic system in state–space form as follows:

$$\begin{cases} x_{t+1} = Ax_t + Bu_t + w_t, \\ y_t = Cx_t + e_t, \end{cases} \tag{1}$$

where $x \in \mathbb{R}^n$ and $u \in \mathbb{R}^m$ denote the plant's state and input vectors, respectively, while $y \in \mathbb{R}^p$ is the plant's output vector obtained from measurements of p sensors from the set $S = \{1, 2, \ldots, p\}$. The process noise $w_t \in \mathbb{R}^n$ and the measurement noise $e_t \in \mathbb{R}^m$ obey some zero-mean stochastic distributions. Moreover, $A \in \mathbb{R}^{nxn}$ is the system matrix, $B \in \mathbb{R}^{nxp}$ is the actuator matrix and $C \in \mathbb{R}^{mxn}$ is the measurement matrix. The next state x_{t+1} depends on the current state x_t and the corresponding control actions u_t, at the sampling instant $t \in N$.

As shown in Fig. 2, the physical plant is supported by a communication network through which the sensor measurements and actuator data are exchanged with the controller. The main interactions between cyber and physical components can be described as follows:

- The interactions between the *physical plant* and *sensors*
- The interaction between the *sensors* and the *controller*
- The interactions between the *controller* and the *actuators*
- The interactions between the *actuators* and the *physical plant*

An applied $\pi-$ calculus representation

The combination of the *physical systems* (G) and the *cyber components* (P) which represents a typical example of CPS, improves the operations of the physical systems but introduces challenges to the bidirectional synchronisation between the components. Based on Lanotte et al. [14] work, a variant of applied $\pi-$calculus is used to formalise and model the interactions of the CPS.

Physical Component: let $\bar{\mathcal{X}} \subseteq \mathcal{X}$ be a set of state variables, $\bar{\mathcal{A}} \subseteq \mathcal{A}$ be a set of actuators, and $\bar{S} \subseteq S$ be a set of sensors. The physical environment G is represented as $\{\xi_x, \xi_u, \xi_w, evol, \xi_e, meas, inv, safe, secure\}$, where:

- $\xi_{\mathcal{X}} \in \mathbb{R}^{\bar{\mathcal{X}}}$ is the *state function* that returns the current value associated to each variable in $\mathcal{X}$
- $\xi_u \in \mathbb{R}^{\bar{\mathcal{A}}}$ is the *actuator function* that returns the current value associated to actuators in $\mathcal{A}$
- $\xi_w \in \mathbb{R}^{\bar{\mathcal{X}}}$ is the *uncertainty function* that returns the uncertainty/accuracy associated to each state variable
- $evol$: $\mathbb{R}^{\bar{\mathcal{X}}} x \mathbb{R}^{\bar{\mathcal{A}}} x \mathbb{R}^{\bar{\mathcal{X}}} \rightarrow 2^{\mathbb{R}^{\bar{\mathcal{X}}}}$ is the *evolution map* that models the evolution law of the physical system, where changes made on the actuators may reflect on state variables
- $\xi_e \in \mathbb{R}^{\bar{S}}$ is the *sensor-error function* that returns maximum error associated to sensors in $\bar{S}$
- $meas$: $\mathbb{R}^{\bar{\mathcal{X}}} x \mathbb{R}^{\bar{S}} \rightarrow 2^{\mathbb{R}^{\bar{S}}}$ is the *measurement map* that returns the set of next admissible sensor measurements based on the current state function
- inv: $\mathbb{R}^{\bar{\mathcal{X}}} \rightarrow \{true, false\}$ is the *invariant set* that returns the set of state functions that satisfy the invariant of the system
- $safe$: $\mathbb{R}^{\bar{\mathcal{X}}} \rightarrow \{true, false\}$ is the *safety function* that represents the set of state functions that satisfy the safety conditions of the system
- $secure$: $\mathbb{R}^{\bar{\mathcal{X}}} \rightarrow \{true, false\}$ is the *security function* that represents the set of state functions that satisfy the security properties of the system. Specifically, if a CPS gets into an insecure state, then its security property may get compromised

The *cyber components* of a CPS are defined using an applied $\pi-$calculus with constructs to read values detected at the sensors and write values on actuators. Processes are defined as follows:

$P, Q ::= nil \quad | \quad \tau.P \quad | \quad P|Q \quad | \quad \lfloor \pi.P \rfloor Q \quad | \quad if$
$(b)\ \{P\} \quad else\ \{Q\}$
$\pi ::= snd\ \bar{c}\langle v \rangle \mid rcv\ c(x) \mid read\ s(x) \mid write\ \bar{a}\langle v \rangle$
$\mu ::= forge\ p\langle v \rangle \quad | \quad drop\ a(x)$

The *nil* represents a *terminated process*. The process $\tau.P$ represents a silent action and then continues as P. $P|Q$ denotes the parallel composition of concurrent threads P and Q. Thus, $\lfloor snd\ \bar{c}\langle v \rangle.P \rfloor Q$ sends the value v on channel c, and it continues as P; otherwise, it evolves into Q. The process $[rcv\ c(x).P]Q$ represents the reception case. The process $[read\ s(x).P]Q$ reads the value detected by the sensor S, whereas $[write\ \bar{a}\langle v \rangle.P]Q$ writes on the actuator a. The process $if(b)\{P\}else\{Q\}$ is the standard conditional, where b is a decidable guard. For $\{\mu \in forge\ p\langle v \rangle, \quad drop\ a(x)\}$, the process $\lfloor \mu.P \rfloor Q$ denotes the threats targeting a CPS system. Specifically, the attacks represent integrity attacks on data coming from sensors to the controller and dropping of actuator commands.

Labelled transition semantics

We define the semantics of the applied π-calculus using a *labelled transition system* to highlight the CPS interactions with the environment and enable the definition of observational equivalences such as *bisimilarity*. The operational semantics is given in Tables 2 and 3. The rules of Table 2 describe the behaviour of processes whereas the rules in Table 3 describe the behaviour of a CPS. A transition of P has the form $P \xrightarrow{\alpha} P'$, specifying that P can perform action α to evolve into P' where α can represent different actions. The meta-variable α ranges over labels in the set $\{nil, \tau, \bar{c}v, cv, a!v, s?v, p!v, p?v,$
$\tau : p\}$. Rules(*outp*), (*Inpp*) and (*Com*) serve to model channel communication on some channel c. Rules (*write*) and (*read*) denote the writing/reading of some data on the physical device p. Rule *(SensWrite)* models an integrity attack on sensor s. Rule (*Par*) propagates untimed actions over parallel components.

The transition rules for the physical and cyber components are given in Table 3. A CPS can evolve if the invariant property is satisfied, otherwise the system will be in undesirable state or deadlocked. Actions ranged over by α are in the set $\{\tau, \bar{c}v, cv, nil\}$. These actions denote non-observable activities (τ), observable activities such as channel transmissions ($\bar{c}v$ and cv). Rules (*out*) and (*Inp*) model transmission and reception with an external system on a channel c. Rule (*SensRead*) models the reading of the current data detected at sensor s. Rule(*ActWrite*) models the writing of a value v on an actuator a.

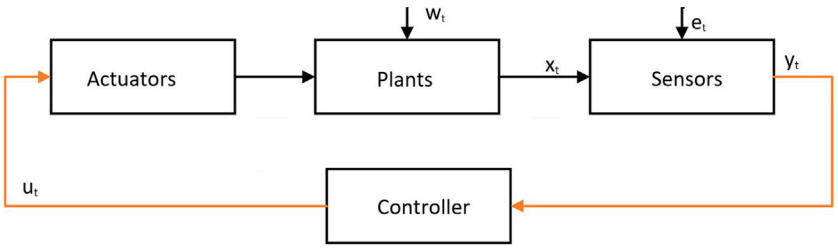

Fig. 2. Interaction of a simplified CPS components.

Table 2
LTS for processes.

$$(Outp)\ \frac{-}{\lfloor snd\, c\langle v\rangle.P\rfloor Q \xrightarrow{\bar{c}v} P} \qquad (Inpp)\ \frac{-}{\lfloor rcv\, c(x).P\rfloor Q \xrightarrow{cv} P\{v/x\}}$$

$$(Com)\ \frac{P \xrightarrow{\bar{c}v} P' \quad Q \xrightarrow{cv} Q'}{P|Q \xrightarrow{\tau} P'|Q'} \qquad (Par)\ \frac{P \xrightarrow{\lambda} P' \quad \lambda \neq nil}{P|Q \xrightarrow{\lambda} P'|Q}$$

$$(Write)\ \frac{-}{\lfloor writea\langle v\rangle.P\rfloor Q \xrightarrow{a!v} P} \qquad (Read)\ \frac{-}{\lfloor reads(x).P\rfloor Q \xrightarrow{s?v} P\{v/x\}}$$

$$(forge)\ \frac{P \xrightarrow{!s!v} P' \quad Q \xrightarrow{!s?v} Q'}{P|Q \xrightarrow{\tau:s} P'|Q'} \qquad (ActRead)\ \frac{P \xrightarrow{a!v} P' \quad Q \xrightarrow{!a?v} Q'}{P|Q \xrightarrow{\tau:a} P'|Q'}$$

Table 3
LTS for CPS.

$$(Out)\ \frac{P \xrightarrow{\bar{c}v} P' \quad inv(G)}{G \bowtie P \xrightarrow{\bar{c}v} G \bowtie P'} \qquad (Inp)\ \frac{P \xrightarrow{cv} P' \quad inv(G)}{G \bowtie P \xrightarrow{cv} G \bowtie P'}$$

$$(SensRead)\ \frac{P \xrightarrow{s?v} P' \quad s\in S \quad inv(G) \quad v\in read\_sensor(G,s)}{G \bowtie P \xrightarrow{\tau} G \bowtie P'} \qquad (Tau)\ \frac{P \xrightarrow{\tau} P' \quad inv(G)}{G \bowtie P \xrightarrow{\tau} G \bowtie P'}$$

$$(ActWrite)\ \frac{P \xrightarrow{a!v} P' \quad a\in S \quad G'=update\_act(G,a,v)}{G \bowtie P \xrightarrow{\tau} G' \bowtie P'} \qquad (Deadlock)\ \frac{'inv(G)}{G \bowtie P \xrightarrow{\text{deadlock}} G \bowtie P}$$

$$(Safety)\ \frac{'safe(G) \quad inv(G)}{G \bowtie P \xrightarrow{\text{unsafe}} G \bowtie P} \qquad (Security)\ \frac{'secure(G) \quad inv(G)}{G \bowtie P \xrightarrow{\text{insecure}} G \bowtie P}$$

Bisimulation

Bisimulation is a binary relation between state transition systems in which the systems behave in the same way in the sense that one system simulates the other and vice versa. The operational semantics of the CPS is described above in terms of a Labelled Transition Semantics similar to the SOS style of Plotkin [38]. This subsection defines a *weak bisimulation*-based behavioural equivalence for CPS. The capability to observe physical events depends on the capability of the cyber components to recognise these events by acting on sensors and actuators, and also, the transmission of messages (over unrestricted channels) can be observed.

Consider the labelled transition system $A = (S, Act, \rightarrow, s, T)$. A relation R over CPSs $R \subseteq SxS$ is defined as a *weak bisimulation relation* iff for all $s, t \in S$ such that $s\ R\ t$, the following conditions hold [39]:

1. If $s \xrightarrow{\alpha} s'$, then
   - either $\alpha = \tau$ and $s'\ R\ t$, or
   - there is a sequence $t \xrightarrow{\tau} \ldots \xrightarrow{\tau} \xrightarrow{a} \xrightarrow{\tau} \ldots \xrightarrow{\tau} t'$ such that $s'\ R\ t'$
2. Symmetrically, if $t \xrightarrow{\alpha} t'$, then
   - either $\alpha = \tau$ and $s\ R\ t'$, or
   - there is a sequence $s \xrightarrow{\tau} \ldots \xrightarrow{\tau} \xrightarrow{\alpha} \xrightarrow{\tau} \ldots \xrightarrow{\tau} s'$ such that $s'\ R\ t'$
3. If $s \in T$, then there is a sequence $t \xrightarrow{\tau} \ldots \xrightarrow{\tau} t'$ such that $t' \in T$
4. Again, symmetrically, if $t \in T$, then there is a sequence $s \xrightarrow{\tau} \ldots \xrightarrow{\tau} s'$ such that $s' \in T$

Two states s, t are weakly bisimilar ($s \approx t$) if there exists a weak bisimulation R such that $\langle s, t\rangle \in R$. In order to consider two states equivalent, it is necessary that for each visible action performed by one of them, the other has to have the possibility of performing the same visible action possibly preceded and followed by any number of invisible actions.

We consider two states (systems) are equivalent if they behave indistinguishably in the presence/absence of any adversary, where the adversary can compromise the security property of the system.

5.2. Threat modelling of cyber–physical systems using an applied π-calculus

This section formalises a threat model using an applied π-calculus to specify denial of service and man-in-the-middle (MITM) attacks that can manipulate sensor readings or control commands in order to drive a CPS into an undesired state. In Fig. 2, the output of the process (y_t) is transmitted over a communication network, and the received output is used to compute control actions (u_t) which are sent back to the physical process. In between the transmission and reception of sensor data and control commands, an attacker may replace the signals coming from the sensors to the controller and from the controller to the actuators. Thus, after each transmission and reception, the attacked output $\bar{y}$ and attacked input $\bar{u}$ take the form

$$\begin{cases} \bar{y}_t := y_t + \delta_t^y, \\ \bar{u}_t := u_t + \delta_t^u, \end{cases} \tag{2}$$

where $\delta_t^y \in \mathbb{R}^m$ and $\delta_t^u \in \mathbb{R}^l$ denote additive sensor and actuator attacks, respectively.

Then, a system under attack from Eqs. (1) is modelled by

$$\begin{cases} x_{t+1} = Ax_t + B(u_t + \delta_t^u) + w_t, \\ \bar{y}_t = Cx_t + e_t + \delta_t^y. \end{cases} \tag{3}$$

A residue vector (Δz_t) represents the difference between the system in the presence of attacks $\bar{z}_t$ and the system without attacks z_t, and it determines if an attack can be detected or not. An attack is hardly detectable if Δz_t is small enough, i.e., there exist $\delta > 0$ such that if $\|\Delta z_t\| \leq \delta, \forall t \in \mathcal{N}$ [40].

Different attack scenarios can also be considered in the architecture illustrated in Fig. 2:

1. An attacker can inject false measurement into the system by faking sensor data and causing the controller to act on malicious data. This can be formalised in the applied calculus as:

   $$S(forge\ M)_{adv} = snd\ c_{SC}\langle M\rangle$$

   where c_{SC} is the channel used by the sensor for sending measurement value to the controller;
2. An attacker may be able to compromise the controller and send incorrect control signals to the actuators. This can be formalised in the applied calculus as:

   $$C = rcv\ c_{SC}(x).\ snd\ c_{CA}\langle comprom.\ x\rangle_{adv}$$

   where c_{SC} is the channel used by the controller for receiving measurement value from the sensor and c_{CA} is the channel used by the controller for sending a control signal to the actuators;

3. An attacker can compromise the actuators and execute a control action that is different to what the controller intended. This can be formalised in the applied calculus as:

$$A = rcv\ c_{CA}(comprom.\ x)_{adv}$$

where c_{CA} is the channel used by the actuators for receiving measurement value from the controller.

Depending on the motive and level of access, an adversary may block and/or modify messages from a compromised device or link. For example, if an attacker controls a sensor that outputs a measurement, then the controller may receive a corrupted version of the measurement. And because we are explicitly modelling distributed systems, we are not able to obtain a global view of the dynamical state of the system. However, we can approximate the dynamics of the system using the local state of the processes. These dynamics will be reflected in the message passing in the applied π-calculus.

Moreover, we are interested in the threats posed by cyber–physical attacks and are concerned with events that leaves a reflection in the cyber domain. Since the cyber domain is a distributed system, we consider the synchronisation that can be observed and modified by the attacker. And for the physical system, the internal processes will evolve according to certain dynamics but what can be observed are only the reflection of the internal dynamics whenever there is an interaction with another process. The zero dynamics or the internal behaviour of the processes is beyond the scope of our model. Also, labelled bisimilarity has been showed to be the same as observation equivalence [12,41]. This implies that as long as we have bisimilarity and can prove it, the internal dynamics of the physical system process can be inferred.

To formally reason about the necessary conditions for the attacks described above to be successful, we use attack–defence trees (ADT) which are employed to analyse an attack–defence scenario [42] and extend it with partial ordering of events to show causality relationship. We then translate the ADT with partial ordering into an applied π-calculus using the message synchronisation primitives for partial ordering. This is because an ADT shows that a particular attack will succeed if some conditions are met but the approach only works if there is no timing or sequencing of events. The abstract representation of ADT with partial ordering is depicted in Fig. 3.

One of the main goals of an attacker in the CPS environment is to cause an undesirable state change in the physical system. We consider two forms of attacks that can be deployed to drive the physical system into an undesirable state: attacking the sensor or attacking the actuator. In order to drive the physical system into an undesirable state using the sensor, the attacker needs both, to compromise the communication channel and to inject false sensor measurement value. We ignore how an attacker might inject the false sensor measurement value and focus on how the communication channel is compromised. The communication channel could be compromised using MITM attack. However, the defender could counter the attacker's action by securing the communication channel. This defence mechanism is subject to the requirements of the specific CPS environment.

For actuator attack, the attacker needs to compromise the communication channel and to inject false control command. We ignore how an attacker might inject the false control command and focus on the communication channel is compromised. Similar to the sensor attack, the communication channel can be compromised through MITM attack. Also, the defender can protect against this attack by securing the communication channel. This protection mechanism would have to be designed so as to meet the requirements of the specific CPS environment that such mechanism would be deployed.

The ADT with partial ordering representing the above described state is shown in Fig. 4. And the ADTerm representing the ADT using the semantics that can be found in [42] with an extension to show partial ordering of events is given as follows:

$$\bigvee^{p}\Big[\wedge^{p}\Big(\text{MITM}, \leq c^{p}\ (\wedge^{p}(\text{Comm. Channel, False Sensor Value}), c^{o}(\text{Secure Comm. Channel})\ ), \leq \text{Sensor}\Big), \wedge^{p}\Big(\text{MITM}, \leq c^{p}\ (\wedge^{p}(\text{Comm. Channel, False Control Command}), c^{o}(\text{Secure Comm. Channel})\ ), \leq \text{Actuator}\Big)\Big]$$

This formalism can be translated into the applied π-calculus using the message synchronisation primitives for partial ordering to represent not only the necessary conditions for an attack to be successful but also the sequencing of events or the way events have to be order for the attack to be successful. This allows us to reason about the timing property of the CPS because the concept of time is derived from the order in which events occur [43]. Although we do not consider explicit timing, we use partial ordering of events to represent dependencies of internal states and the function over these states and how they are linked together with the message — basically the semantics of the applied π-calculus. Thus, we have internal states that represent the current states and then we have a message and that message is implicitly creating a partial ordering over the events which we might consider the equivalent of ADT but with sequencing of operation.

To translate the ADT with partial ordering in Fig. 4 into an applied π-calculus using the message synchronisation primitives for partial ordering and taking the sensor attack into consideration, the sensor (S), controller (C) and actuator (A) activities can be represented as parallel composition ($S|C|A$). This composition where S and C are connected by a channel c_{SC}, and C and A by a channel c_{CA} shows the partial ordering of the events within the CPS. The sensor uses c_{SC} channel for sending a measurement to the controller, and the controller uses c_{CA} channel for sending a control signal to the actuators. We can represent these partial orders as follows:

$$S \rightarrow C: \quad M \quad on \quad channel \quad c_{SC}$$
$$C \rightarrow A: \quad M \quad on \quad channel \quad c_{CA}$$

The actual applied $\pi-$calculus description of this message interaction (M) is:

$$S(M) = snd\ c_{SC}\langle M\rangle$$
$$C = rcv\ c_{SC}(x).\ snd\ c_{CA}\langle x\rangle$$
$$A = rcv\ c_{CA}(x)$$
$$Ctrl(M) = (\nu c_{SC})(\nu c_{CA})(S(M)|C|A)$$

Thus, the whole CPS is defined as: $CPS = G \bowtie Ctrl(M)$, where G is the physical environment defined in Section 5.1.

Similarly, the adversary's actions must coincide with message transactions for an attack to be successful. We consider Dolev–Yao threat model [44] where an adversary can compromise the communication channel to inject false measurement into the system by faking sensor data and causing the controller to act on malicious data. An applied π-calculus description of the false measurement injection using the compromised communication channel and false sensor measurement value S_{adv} can be given as follows:

$$S(forge\ M)_{adv} = snd\ c_{SC}\langle M\rangle$$
$$C = rcv\ c_{SC}(x).\ snd\ c_{CA}\langle x\rangle$$
$$A = rcv\ c_{CA}(x)$$
$$Ctrl(M)_{adv} = (\nu c_{SC})(\nu c_{CA})(S(M)_{adv}|C|A)$$

Thus, the CPS under attack is defined as:

$$C\bar{P}S = G \bowtie Ctrl(M)_{adv}$$

It is then trivial to derive the necessary conditions under which the attack will be successful and the way events have to be order for the

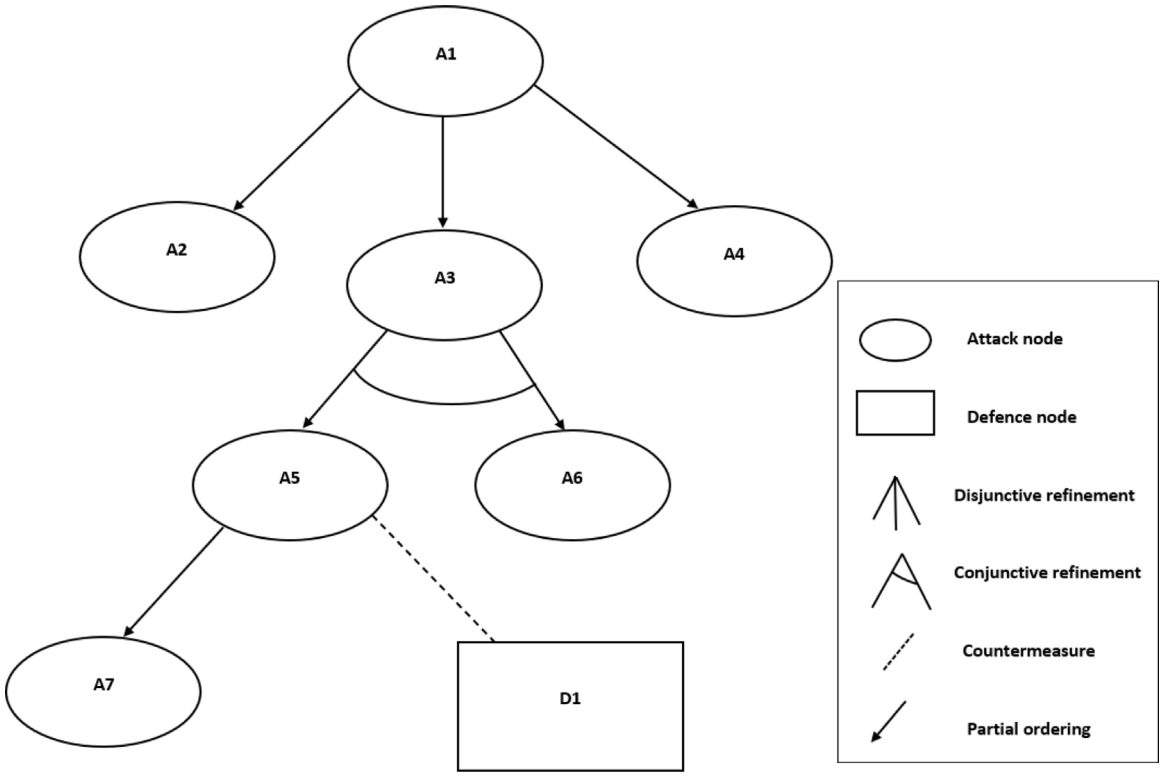

Fig. 3. An abstract attack–defence tree with partial ordering.

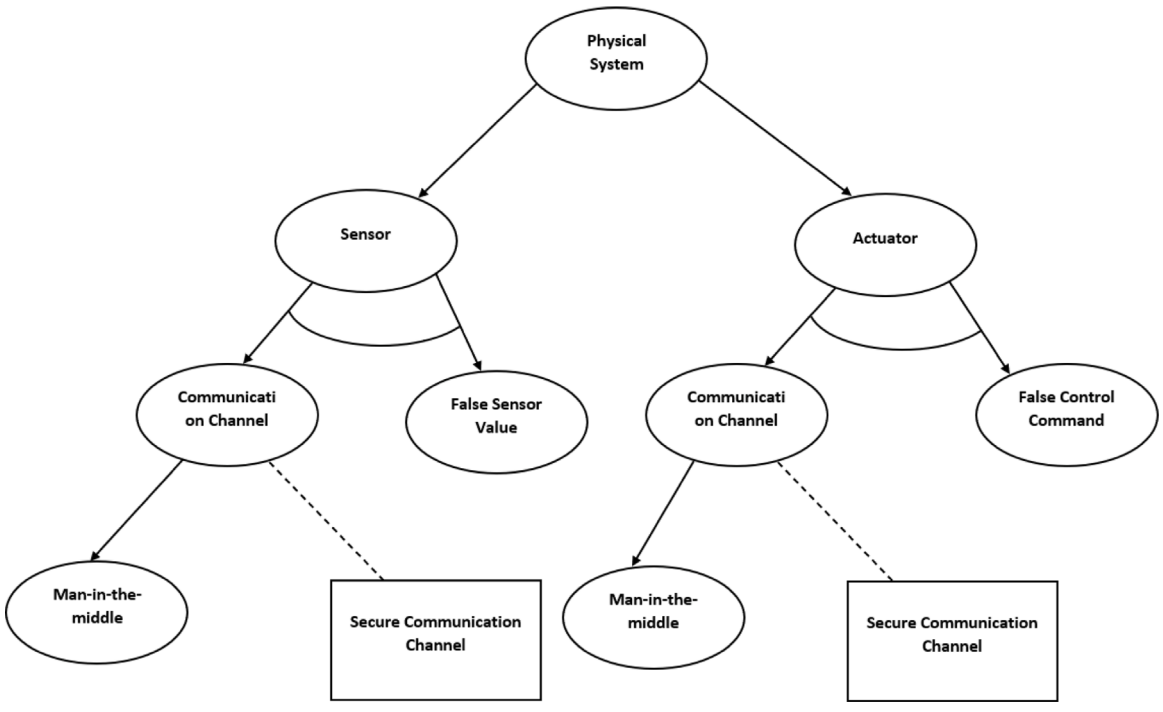

Fig. 4. The attack–defence tree with partial ordering for CPS attack.

attack to be successful. Therefore, our threat model allows us to know not only the conditions the attacker would have to meet to be successful but also the partial ordering of the events needed for the successful execution of the attack in the CPS environment.

5.3. Use case scenario

In this subsection, we demonstrate the utility of our proposed model using the IEC 61850 based substation. The IEC 61850 is a standard

which defines the communication requirements for substation automation [45]. The communication architecture for substation automation as defined by IEC 61850 standard include three different levels of communication, namely: the station level, the bay level and the process level [45]. The IEC 61850 communication architecture is shown in Fig. 5.

However, for the purpose of illustrating the use of our model, we use an applied π-calculus to reason about the potential threats to the interaction within/between the process level and the bay level. The process level as can be seen from Fig. 5 consists of devices such intelligent switchgear, current transformer (CT), and voltage transformer (VT); and the bay level consists of protection and control devices which include relays, intelligent electronic devices (IEDs), and phasor measurement unit (PMU). IEC 61850 protocols such as the generic object-oriented substation event (GOOSE) and sampled value (SV) are then employed for communication within/between the process level and the bay level. Whilst GOOSE is used to send tripping signals from the IEDs to a circuit breaker (CB), SV is used to send sampled voltage and current values from MU to IEDs.

The bay level consists of different IEDs that collect sample measured values from the process level devices (currents and voltages using sampled value messages from merging units) via the communication channel (local area network). The IEDs can make local control decisions, protect irregularities from the process level, transmit data to other IEDs, or send the data to the substation level for further processing and monitoring. Also, the CB will receive a trip signal from the IEDs using a GOOSE message travelling in the process bus.

Moreover, there are situations where the CB may fail. The protective mechanisms for such scenarios are initiated via the communication channel. The IED relay would broadcast a GOOSE message to the adjacent break control relays. On receiving the GOOSE message, the breaker control relays would trip and block close their respective breakers. The breaker control relays also communicate with each other using the communication channel.

However, an adversary may inject false streaming measurement data (currents and voltages) with the intent of causing the protective relays to issue false tripping commands (inducing the IED to trip CBs). For instance, a malicious MU (sensor) can issue a false SV to an IED (controller) that indicates a fault current when there is no fault, and it leads to a needless CB trip action by the relay subscribing to the SV stream.

The interaction between the adversary and the defender can be represented using the ADT with partial ordering. As we have observed in the description of our use case scenario, the main goal of the attacker is to cause physical state change i.e., needless CB trip action by the relay subscribing to the SV stream. To achieve this goal, the attacker would have to cause a sensor change. This sensor change can be triggered by compromising the communication channel and injecting false SV value. In this type of attack, MITM approach can be adopted by in order to compromise the communication channel. The defender, on the other hand, can counter the actions of the attacker by securing the communication channel between the sensor and the controller. The ADT with partial ordering showing this interaction between the attacker and the defender for the use case scenario we have described is equivalent to the sensor attack path of the ADT with partial ordering for CPS attack in Section 5.2.

The ADT translation into an applied π-calculus using the message synchronisation primitives for partial ordering, representing the MU, IED and CB regular measurement (RM) interaction instance, and its variant with the false measurement injection (FMI) attack is given as follows:

$$\begin{aligned} S(M) &= snd\ c_{SC}\langle M\rangle \\ C &= rcv\ c_{SC}(x).\ snd\ c_{CA}\langle x\rangle \\ A &= rcv\ c_{CA}(x) \\ RM(M) &= (\nu c_{SC})(\nu c_{CA})(S(M)|C|A) \end{aligned}$$

The MU measurement instance with the false measurement injection attacks is given as follows:

$$\begin{aligned} S(forge\ M)_{adv} &= snd\ c_{SC}\langle M\rangle \\ C &= rcv\ c_{SC}(x).\ snd\ c_{CA}\langle x\rangle \\ A &= rcv\ c_{CA}(x) \\ FMI(M)_{adv} &= (\nu c_{SC})(\nu c_{CA})(S(forge\ M)_{adv}|C|A) \end{aligned}$$

We can then make an argument for equivalence where we can say whether that this attack can be successful. For example, the outputs of these two instances (i.e., $RM(M)$ and $FMI(M)_{adv}$) can be considered indistinguishable by an external observer such as the defence mechanism. This *undetectability* property can be formalised as *observational equivalence* [10]. Observational equivalence is used to express the notion that two system instances are observationally equivalent if they behave indistinguishably *from the defender's or the attacker's perspective*. It can be used to specify security properties such as the inability of the defender or the attacker to distinguish between two instances of a system. To show the undetectability property of the two output instances, we use weak bisimulation defined in Section 5.1 as follows.

Given the two instances $RM(M)$ and $FMI(M)_{adv}$, let us assume that

$$P_i \approx RM(M)$$

and

$$Q_i \approx FMI(M)_{adv}$$

for all i. Our goal is to show that $P_i \approx Q_i$ for each i. To achieve this, it is sufficient to demonstrate that

$$S = \{\ (P, Q)\ |\ P \approx P_i \text{ and } Q \approx Q_i \text{ for some } i\ \}$$

is a weak bisimulation.

Proof. So, let us consider an arbitrary pair $(P, Q) \in S$. First, suppose $P \Rightarrow P'$. Then

$$P \approx P_i \approx RM(M) \Rightarrow P'' \approx P', \text{ for some } P''.$$

But since $\alpha_{ij} \neq \tau$ for all i, j; it follows that $P \approx P'$. By selecting $Q' = Q$, we actually have a Q' such that $Q \Rightarrow Q'$ and $(P', Q') \in S$.

Also, suppose $P \overset{\lambda}{\Rightarrow} P'$. Then

$$P \approx P_i \approx RM(M) \overset{\lambda}{\longrightarrow} P_k \Rightarrow P'' \approx P',$$

where $\lambda = \alpha_{ij}$ and $P_k = P_{k(ij)}$ for some j. Using the same reasoning as above, $P_k \approx P'$. Further, we have

$$FMI(M)_{adv} \overset{\lambda}{\longrightarrow} Q_k;$$

but

$$Q_i \approx FMI(M)_{adv},$$

so

$$Q \overset{\lambda}{\Rightarrow} Q' \approx Q_k$$

for some Q', which is what we require to complete the proof.

Consequently, for the false measurement injection attack we have described, the necessary conditions for the attack to be successful and the sequencing of events or the way events have to be order for the attack to be successful are given by the sensor attack path of the ADT with partial ordering for CPS attack in Fig. 4. We also translated the ADT with partial ordering into an applied $\pi-$calculus using the message synchronisation primitives for partial ordering so as to make an argument for equivalence; where we have showed that the false measurement injection attack can be successful if the output of the two instances – $RM(M)$ and $FMI(M)_{adv}$ – are observationally equivalent.

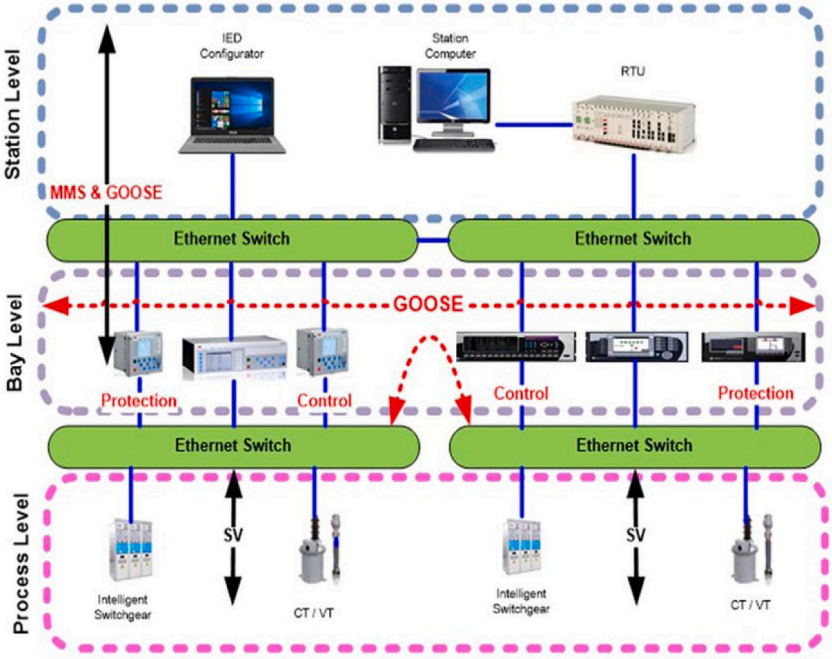

Fig. 5. IEC 61850 communication architecture [46].

6. Conclusion and future work

Indeed, an understanding of assets that make up a CPS and the use of an applied π-calculus can provide useful insights about threats to a CPS and help in reasoning about its behaviour. We have presented threat modelling of CPS using an applied π-calculus in this paper. We argued that the regular threat modelling approaches are not suitable for capturing the threats to CPS considering the uncertainty, timing and dependencies that exist between its entities. We then proposed an extension to an applied π-calculus which allows us to capture both the behaviour of the CPS as well as modelling possible adversary behaviour. Lastly, the utility of our model was demonstrated for the case of an electrical substation fragment in which components communicate via the IEC 61850 protocol.

In the future, we hope to consider the possibility of automating the threat modelling process of CPS using the theoretical foundations presented in this work. This is because the use of an automated support tool can facilitate the threat modelling of a much more complex system. In addition, we intend to expand our approach to investigate threats in other sectors within the critical infrastructure. To that end, an adequate applied π-calculus model for the system under consideration would have to be developed.

Declaration of competing interest

No author associated with this paper has disclosed any potential or pertinent conflicts which may be perceived to have impending conflict with this work. For full disclosure statements refer to https://doi.org/10.1016/j.ijcip.2021.100466.

References

[1] K. Zetter, An unprecedented look at stuxnet, the world's first digital weapon, 2014, https://www.wired.com/2014/11/countdown-to-zero-day-stuxnet/.
[2] D. Palmer, Ransomware attacks are now targeting industrial control systems, 2020, https://www.zdnet.com/article/ransomware-attacks-are-now-targeting-industrial-control-systems/.
[3] NIST, Information security: Guide for conducting risk assessments, 2012, https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-30r1.pdf.
[4] A. Shostack, Threat Modeling: Designing for Security, John Wiley & Sons, 2014.
[5] T. UcedaVelez, M.M. Morana, Risk Centric Threat Modeling: Process for Attack Simulation and Threat Analysis, John Wiley & Sons, 2015.
[6] C. Alberts, A. Dorofee, J. Stevens, C. Woody, Introduction to the OCTAVE approach, 2003, http://dx.doi.org/10.21236/ada634134.
[7] E.B. Fernandez, Threat modeling in cyber-physical systems, in: Proc. Nd Intl Conf Big Data Intelligence and Computing and Cyber Science and Technology Congress(DASC/PiCom/DataCom/CyberSciTech) 2016 IEEE 14th Intl Conf Dependable, Autonomic and Secure Computing, 14th Intl Conf Pervasive Intelligence and Computing, 2016, pp. 448–453, http://dx.doi.org/10.1109/DASC-PICom-DataCom-CyberSciTec.2016.89.
[8] M.A.H. Sadi, M.H. Ali, D. Dasgupta, R.K. Abercrombie, S. Kher, Co-simulation platform for characterizing cyber attacks in cyber physical systems, in: Proc. IEEE Symp. Series Computational Intelligence, 2015, pp. 1244–1251, http://dx.doi.org/10.1109/SSCI.2015.178.
[9] R. Milner, J. Parrow, D. Walker, A calculus of mobile processes, i, Inform. and Comput. 100 (1) (1992) 1–40, http://dx.doi.org/10.1016/0890-5401(92)90008-4.
[10] D. Sangiorgi, D. Walker, The Pi-Calculus: A Theory of Mobile Processes, Cambridge university press, 2003.
[11] M. Abadi, A.D. Gordon, A calculus for cryptographic protocols: The spi calculus, Inform. and Comput. 148 (1) (1999) 1–70, http://dx.doi.org/10.1006/inco.1998.2740.
[12] M. Abadi, C. Fournet, Mobile values, new names, and secure communication, ACM SIGPLAN Notices 36 (3) (2001) 104–115, http://dx.doi.org/10.1145/373243.360213.
[13] S. Kremer, M. Ryan, Analysis of an electronic voting protocol in the applied pi calculus, in: Programming Languages and Systems, Springer Berlin Heidelberg, 2005, pp. 186–200, http://dx.doi.org/10.1007/978-3-540-31987-0_14.
[14] R. Lanotte, M. Merro, A calculus of cyber-physical systems, in: F. Drewes, C. Martín-Vide, B. Truthe (Eds.), Language and Automata Theory and Applications - 11th International Conference, LATA 2017, Umeå, Sweden, March 6-9, 2017, Proceedings, in: Lecture Notes in Computer Science, 10168, 2017, pp. 115–127, http://dx.doi.org/10.1007/978-3-319-53733-7_8.
[15] J. Parrow, An introduction to the pi-calculus, in: Handbook of Process Algebra, Elsevier, 2001, pp. 479–543, http://dx.doi.org/10.1016/b978-044482830-9/50026-6.
[16] Y. Mo, R. Chabukswar, B. Sinopoli, Detecting integrity attacks on SCADA systems, IEEE Trans. Control Syst. Technol. 22 (2014) 1396–1407, http://dx.doi.org/10.1109/TCST.2013.2280899.
[17] F. Pasqualetti, F. Dorfler, F. Bullo, Attack detection and identification in cyber-physical systems, IEEE Trans. Autom. Control 58 (11) (2013) 2715–2729, http://dx.doi.org/10.1109/tac.2013.2266831.
[18] S. Amin, A.A. Cárdenas, S.S. Sastry, Safe and secure networked control systems under denial-of-service attacks, in: Hybrid Systems: Computation and Control, Springer Berlin Heidelberg, 2009, pp. 31–45, http://dx.doi.org/10.1007/978-3-642-00602-9_3.

[19] M. Doostmohammadian, U.A. Khanc, Vulnerability of CPS inference to dos attacks, in: 2014 48th Asilomar Conference on Signals, Systems and Computers, IEEE, 2014, http://dx.doi.org/10.1109/acssc.2014.7094825.

[20] Y. Mo, B. Sinopoli, Secure control against replay attacks, in: 2009 47th Annual Allerton Conference on Communication, Control, and Computing (Allerton), IEEE, 2009, http://dx.doi.org/10.1109/allerton.2009.5394956.

[21] H.S. Sánchez, D. Rotondo, T. Escobet, V. Puig, J. Saludes, J. Quevedo, Detection of replay attacks in cyber-physical systems using a frequency-based signature, J. Franklin Inst. B 356 (5) (2019) 2798–2824, http://dx.doi.org/10.1016/j.jfranklin.2019.01.005.

[22] L.O. Nweke, G.K. Weldehawaryat, S.D. Wolthusen, Adversary model for attacks against IEC 61850 real-time communication protocols, in: 2020 16th International Conference on the Design of Reliable Communication Networks DRCN 2020, IEEE, 2020, http://dx.doi.org/10.1109/drcn48652.2020.1570604241.

[23] Y. Mo, E. Garone, A. Casavola, B. Sinopoli, False data injection attacks against state estimation in wireless sensor networks, in: 49th IEEE Conference on Decision and Control (CDC), IEEE, 2010, http://dx.doi.org/10.1109/cdc.2010.5718158.

[24] O.A. Beg, T.T. Johnson, A. Davoudi, Detection of false-data injection attacks in cyber-physical DC microgrids, IEEE Trans. Ind. Inform. 13 (5) (2017) 2693–2703, http://dx.doi.org/10.1109/tii.2017.2656905.

[25] B. Chen, H. Li, B. Zhou, Real-time identification of false data injection attacks: A novel dynamic-static parallel state estimation based mechanism, IEEE Access 7 (2019) 95812–95824, http://dx.doi.org/10.1109/access.2019.2929785.

[26] J. Zalewski, S. Drager, W. McKeever, A.J. Kornecki, Threat modeling for security assessment in cyberphysical systems, in: F.T. Sheldon, A. Giani, A.W. Krings, R.K. Abercrombie (Eds.), Cyber Security and Information Intelligence, CSIIRW '13, Oak Ridge, TN, USA, January 8-10, 2013, ACM, 2013, p. 10, http://dx.doi.org/10.1145/2459976.2459987.

[27] G. Martins, S. Bhatia, X. Koutsoukos, K. Stouffer, C. Tang, R. Candell, Towards a systematic threat modeling approach for cyber-physical systems, in: Proc. Resilience Week (RWS), 2015, pp. 1–6, http://dx.doi.org/10.1109/RWEEK.2015.7287428.

[28] R. Khan, K. McLaughlin, D. Laverty, S. Sezer, STRIDE-based threat modeling for cyber-physical systems, in: Proc. IEEE PES Innovative Smart Grid Technologies Conf. Europe (ISGT-Europe), 2017, pp. 1–6, http://dx.doi.org/10.1109/ISGTEurope.2017.8260283.

[29] H. Almohri, L. Cheng, D. Yao, H. Alemzadeh, On threat modeling and mitigation of medical cyber-physical systems, in: Proc. Systems and Engineering Technologies (CHASE) 2017 IEEE/ACM Int. Conf. Connected Health: Applications, 2017, pp. 114–119, http://dx.doi.org/10.1109/CHASE.2017.69.

[30] M. Rekik, C. Gransart, M. Berbineau, Cyber-physical threats and vulnerabilities analysis for train control and monitoring systems, in: Proc. Computers and Communications (ISNCC) 2018 Int. Symp. Networks, 2018, pp. 1–6, http://dx.doi.org/10.1109/ISNCC.2018.8531005.

[31] Y. Atif, Y. Jiang, D. Jianguo, M. Jeusfeld, B. Lindström, S. Andler, C. Brax, D. Haglund, B. Lindström, Cyber-threat analysis for Cyber-Physical Systems, University of Skövde, 2018.

[32] M. Rocchetto, N.O. Tippenhauer, On attacker models and profiles for cyber-physical systems, in: Computer Security – ESORICS 2016, Springer International Publishing, 2016, pp. 427–449, http://dx.doi.org/10.1007/978-3-319-45741-3_22.

[33] S. Adepu, A. Mathur, Generalized attacker and attack models for cyber physical systems, in: 2016 IEEE 40th Annual Computer Software and Applications Conference (COMPSAC), IEEE, 2016, http://dx.doi.org/10.1109/compsac.2016.122.

[34] S. Adepu, A. Mathur, An investigation into the response of a water treatment system to cyber attacks, in: 2016 IEEE 17th International Symposium on High Assurance Systems Engineering (HASE), IEEE, 2016, http://dx.doi.org/10.1109/hase.2016.14.

[35] L.O. Nweke, S.D. Wolthusen, A review of asset-centric threat modelling approaches, Int. J. Adv. Comput. Sci. Appl. (ISSN: 2156-5570) 11 (2) (2020) 1–6, http://dx.doi.org/10.14569/ijacsa.2020.0110201.

[36] M. Krotofil, A.A. Cárdenas, J. Larsen, D. Gollmann, Vulnerabilities of cyber-physical systems to stale data - determining the optimal time to launch attacks, Int. J. Crit. Infrastruct. Prot. 7 (4) (2014) 213–232, http://dx.doi.org/10.1016/j.ijcip.2014.10.003.

[37] J. Hughes, G. Cybenko, Three tenets for secure cyber-physical system design and assessment, in: Cyber Sensing 2014, Vol. 9097, International Society for Optics and Photonics, 2014, p. 90970A.

[38] G.D. Plotkin, A Structural Approach to Operational Semantics, Computer Science Department, Aarhus University Aarhus, Denmark, 1981.

[39] R. Milner, A Calculus of Communicating Systems, Springer-Verlag Berlin Heidelberg, 1980, p. 174.

[40] G. Wu, J. Sun, J. Chen, Optimal data injection attacks in cyber-physical systems, IEEE Trans. Cybern. 48 (12) (2018) 3302–3312, http://dx.doi.org/10.1109/TCYB.2018.2846365.

[41] M. Arapinis, J. Liu, E. Ritter, M. Ryan, Stateful applied pi calculus, in: Lecture Notes in Computer Science, Springer Berlin Heidelberg, 2014, pp. 22–41, http://dx.doi.org/10.1007/978-3-642-54792-8_2.

[42] B. Kordy, S. Mauw, S. Radomirović, P. Schweitzer, Foundations of attack–defense trees, in: Lecture Notes in Computer Science, Springer Berlin Heidelberg, 2011, pp. 80–95, http://dx.doi.org/10.1007/978-3-642-19751-2_6.

[43] L. Lamport, Time, clocks, and the ordering of events in a distributed system, Commun. ACM 21 (7) (1978) 558–565, http://dx.doi.org/10.1145/359545.359563.

[44] D. Dolev, A. Yao, On the security of public key protocols, IEEE Trans. Inform. Theory 29 (2) (1983) 198–208.

[45] International Electrotechnical Commission (IEC), Communication Networks and Systems in Substations, IEC 61850 Standard, The International Electrotechnical Commission (IEC), 2011.

[46] J. Claveria, A. Kalam, GOOSE Protocol: Ied's smart solution for victoria university zone substation (VUZS) simulator based on IEC61850 standard, in: Proc. IEEE PES Asia-Pacific Power and Energy Engineering Conf. (APPEEC), 2018, pp. 730–735, http://dx.doi.org/10.1109/APPEEC.2018.8566413.